

CLAIMS

What is claimed is:

1

2 1. A method for communicating across a heterogeneous network having
3 components with dissimilar data structure definitions, comprising:

4 prefixing an encoded data structure with a length value that reflects the size of the
5 encoded data structure.

6 2. The method of Claim 1, further comprising decoding the encoded data
7 structure, a down-level receiver reads the length value and decoding the encoded data
8 structure according to the receiver's data definition.

9 3. The method of Claim 2, further comprising, upon completion of decoding,
10 the receiver determining the amount of the encoded data structure that was decoded and
11 skipping the remainder of the encoded data structure according to the length value.

12 4. The method of Claim 3, wherein the method is implemented through
13 instructions on a computer-readable medium, for communicating data between programs
14 along a data communication path.

15 5. The method of Claim 1, further comprising, for a down-level sender, if the
16 up-level definition of a structure has more data elements than the down-level definition of
17 the structure, for built-in type data fields, automatically assigning a default value to any
18 field for which the received data has provided no value, and, for derived type data fields,
19 calling an initialization routine which assigns a default value to any built-in type data
20 field or calls the initialization routine for a derived type data field.

21 6. The method of Claim 5, wherein the method is implemented through
22 instructions on a computer-readable medium, for communicating data between programs
23 along a data communication path.

1 7. A method for communicating across a heterogeneous network having
2 components with dissimilar data structure definitions, comprising:

3 for an up-level sender, if the up-level definition of a structure has more data
4 elements than the down-level definition of the structure, then prefixing an encoded data
5 structure with a length value that reflects the size of the encoded data structure and
6 decoding the encoded data structure in which a down-level receiver reads the length
7 value and decoding the encoded data structure according to the receiver's data definition
8 and upon completion of decoding, the receiver determining the amount of the encoded
9 data structure that was decoded and skipping the remainder of the encoded data structure
10 according to the length value;

11 for a down-level sender, if the up-level definition of a structure has data elements
12 than the down-level definition of the structure, for built-in type data fields, automatically
13 assigning a default value to any field for which the received data has provided no value,
14 and, for derived type data fields, calling an initialization routine which assigns a default
15 value to any built-in type data field or calls the initialization routine for a derived type
16 data field; and

17 if the up-level definition of a structure requires more data elements than the
18 down-level definition of data elements, then following a set of predetermined rules which
19 include extending only data structures which are passed from a server to a client;
20 ensuring that a down-level client's ignorance of extended data causes no ill effects in that
21 client's operational behavior; allowing down-level clients to interact with up-level servers
22 and disallowing up-level clients from interacting with down-level servers; and, in cases
23 where extensions are needed for data structures passed from a client to a server, defining
24 a new data structure that includes both old data fields and new data fields.

25 8. The method of Claim 7, wherein the method is implemented through
26 instructions on a computer-readable medium, for communicating data between programs
27 along a data communication path.

28 9. The method of Claim 8, wherein the method is practiced without creating
29 new data structures.

- 1 10. The method of Claim 8, wherein the method does not use a lock-step
2 migration strategy.
- 3 11. The method of Claim 8, wherein the built-in type includes at least one of
4 the group consisting of integer, floating point, Boolean, and string.
- 5 12. The method of Claim 8, wherein the derived type includes at least one of
6 structure and union.
- 7 13. A system for communicating over a heterogeneous network, comprising:
8 a server; and
9 a client, wherein the server and the client communicate with each other using the
10 method of Claim 8.
- 11 14. The method of Claim 8, wherein the method is implemented using a
12 procedure calling model for distributed applications and a standard representation for
13 data in the network to support heterogeneous network.
- 14 15. The method of Claim 14, wherein the procedure calling model is defined
15 by the Remote Procedure Call (RPC) package and the standard representation of data is
16 accomplished through the External Data Representation (XDR).

- 1 16. A method for communication over a heterogeneous network between an
2 up-level component having an up-level data definition and a down-level component
3 having a down-level data definition, comprising:
4 encoding data in which the sender of the encoded data is either up-level or down-
5 level from the receiver;
6 receiving the encoded data from the sender by the receiver; and
7 decoding the encoded data based on the relationship as to whether the sender is
8 up-level or down-level relative to the receiver and on the relationship of the sender's data
9 definition being greater in size or lesser in size than the receiver's data definition.
- 10 17. The method of Claim 16, wherein the receiver will assign default values or
11 skip data from the encoded data.
- 12 18. The method of Claim 17, wherein the sender attaches a length value to the
13 encoded data.